Amendment to the Claims:

This listing of claims replaces all prior versions, and listings, of claims in the application:

1. (Currently amended) A method comprising:

selecting one of a plurality of debugging modes as a function of a current operating mode of a processor; and

invoking one of a plurality of debug handlers, wherein the plurality of debug handlers includes a first debug handler and a second debug handler, and wherein the first debug handler comprises an emulation service routine and wherein the second debug handler comprises an exception handler.

2. (canceled)

3. (canceled)

4. (Currently amended) The method of claim 1 wherein selecting the debugging mode comprises selecting a first debugging mode when the operating mode comprises [[user]] a supervisor mode, and selecting a second debugging mode when the operating mode comprises ~~supervisor~~ a user mode.

5. (Currently amended) A method comprising:

receiving an instruction;

receiving a signal;

selecting a mode of debugging as a function of the signal, wherein selecting the debugging mode comprises selecting a first debugging mode when the signal is a first signal, [[and]] selecting a second debugging mode when the signal is a second signal, and selecting the debugging mode as a function of a current operating mode of a processor when the signal is a third signal;

invoking one of a plurality of debug handlers, wherein the plurality of debug handlers includes a first debug handler and a second debug handler; and

executing the instruction.

6. (Original) The method of claim 5 further comprising raising an exception.

7. (Original) The method of claim 5 further comprising invoking an emulation event.

8. (Original) The method of claim 5 further comprising:

sensing register contents; and

outputting register contents.

9. (Currently amended) The method of claim 5, wherein the instruction is received by [[a]] the processor, and wherein the processor is adapted to operate in a plurality of states, the method further comprising:

sensing states of the processor; and

outputting states of the processor.

10. (Currently amended) The method of claim 5, wherein: ~~the instruction is received by a processor, the method further comprising selecting a mode of single step debugging as a function of the operating mode of the processor~~

the first debugging mode comprises invoking an emulation event;

the second debugging mode comprises raising an exception; and

the selecting the debugging mode as a function of a current operating mode of a processor comprises selecting the first debugging mode when a current operating mode of the processor comprises a supervisor mode and selecting the second debugging mode when a current operating mode of the processor comprises a user mode.

11. (Currently amended) A device comprising:

a processor, the processor adapted to operate in a plurality of operating modes including an emulation mode;

a control register adapted to store the state of [[a]] at least one control bit; and

a plurality of debug handlers, wherein the plurality of debug handlers includes a first debug handler and a second debug handler, and wherein the first debug handler comprises an emulation service routine and the second debug handler comprises an exception handler;

wherein the processor is adapted to select one of a plurality of debugging modes as a function of the at least one control bit.

12. (Original) The device of claim 11, wherein the processor is adapted to select one of a plurality of debugging modes as a function of the current operating mode of the processor.

13. (Currently amended) The device of claim 11, further comprising exception logic adapted to sense the state of the at least one control bit and to trigger an exception event as a function of the state of the at least one control bit.

14. (Currently amended) The device of claim 11, further comprising emulation logic adapted to sense the state of the at least one control bit and to trigger an emulation event as a function of the state of the at least one control bit.

15. (Currently amended) The device of claim 11, wherein the at least one control bit is a first control bit, the system further comprising a second control register adapted to store the state of a second control bit, and wherein the mode of single-step debugging is a function of the state of the second control bit.

16. (Original) The device of claim 11, wherein the processor is a digital signal processor.

17. (Currently amended) A device comprising:

a processor, the processor adapted to operate in a plurality of operating modes; and

a register adapted to store the state of a signal;

wherein the processor is adapted to select a debugging mode as a function of the signal, wherein selecting the debugging mode comprises selecting a first debugging mode when the signal

is a first signal, selecting a second debugging mode when the signal is a second signal, and selecting the debugging mode as a function of a current operating mode of the processor when the signal is a third signal; and

~~one of a plurality of debugging modes as a function of the current operating mode of the processor and~~ wherein the processor is further adapted to invoke one of a plurality of debug handlers, wherein the plurality of debug handlers includes a first debug handler and a second debug handler.

18. (Original) The device of claim 17 further comprising a control register adapted to store the state of a control bit, wherein the processor is adapted to select one of the plurality of debugging modes as a function of the state of the control bit.

19. (Original) The device of claim 18, further comprising:

an exception handler; and

logic adapted to sense the state of the control bit and to trigger an exception event as a function of the state of the control bit.

20. (Original) The device of claim 18, further comprising logic adapted to sense the state of the control bit and to trigger an emulation event as a function of the state of the control bit.

21. (Original) The device of claim 17, wherein the processor is a digital signal processor.

22. (canceled)

23. (canceled)

24. (canceled)

25. (canceled)

26. (canceled)

27. (canceled)

28. (Previously Presented) The method of claim 1, wherein the first debug handler is capable of debugging the second debug handler.

29. (canceled)

30. (canceled)

31. (Previously Presented) The method of claim 1, further comprising using the first debug handler to debug the second debug handler.

32. (New) The method of claim 1, further comprising determining whether to bypass said selecting one of a plurality of debugging modes as a function of a current operating mode of the processor and to instead select one of a plurality of debug modes without regard to the current operating mode of the processor.

33. (New) The method of claim 5, wherein the selected debug mode comprises raising an exception after executing an instruction, and wherein the current operating mode of the processor comprises a supervisor mode.

34. (New) The device of claim 12, wherein the processor is further adapted to select a first debugging mode when the

current operating mode comprises a supervisor mode and to select a second debugging mode when the current operating mode comprises a user mode.

35. (New) The device of claim 34, wherein the first debugging mode comprises an emulation debugging mode and wherein the second debugging mode comprises an exception debugging mode.

36. (New) The device of claim 17, wherein:

the first debugging mode comprises invoking an emulation event;

the second debugging mode comprises raising an exception; and

said selecting the debugging mode as a function of a current operating mode of a processor comprises selecting the first debugging mode when a current operating mode of the processor comprises a supervisor mode and selecting the second debugging mode when a current operating mode of the processor comprises a user mode.

37. (New) A method comprising:

receiving an instruction;

receiving a signal; and

selecting a debugging mode, wherein said selecting a debugging mode comprises:

selecting an emulation debugging mode when the signal is a first signal;

selecting an exception debugging mode when the signal is a second signal;

selecting the emulation debugging mode when the signal is a third signal and when a current operating mode of a processor comprises a supervisor mode; and

selecting the exception debugging mode when the signal is the third signal and when the current operating mode of the processor comprises a user mode.

38. (New) The method of claim 37, wherein the selected debugging mode comprises an exception debugging mode and wherein the current operating mode of the processor comprises a supervisor mode.

39. (New) The method of claim 37, wherein the third signal signifies that the selected debugging mode should be one level higher than the current operating mode of the processor.